

Workshop git

150 Jahre
culture of
excellence

TUM

Fabio Gratl

Technical University of Munich

Faculty of Informatics

Chair of Scientific Computing in Computer Science (SCCS)

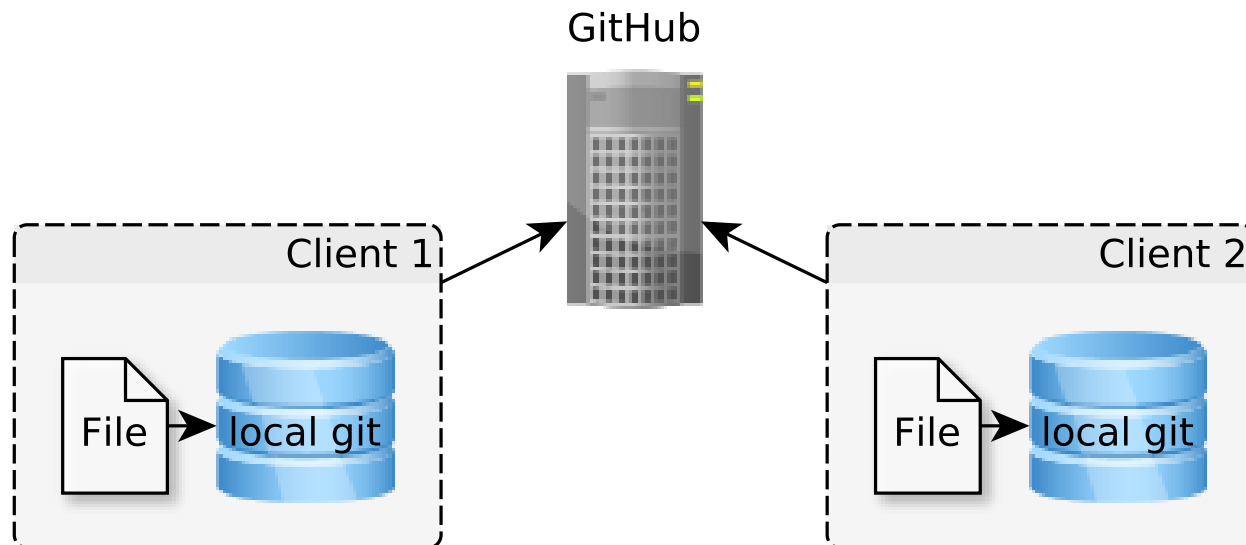
TU Kaiserslautern, 12.10.2018



TUM Uhrenturm

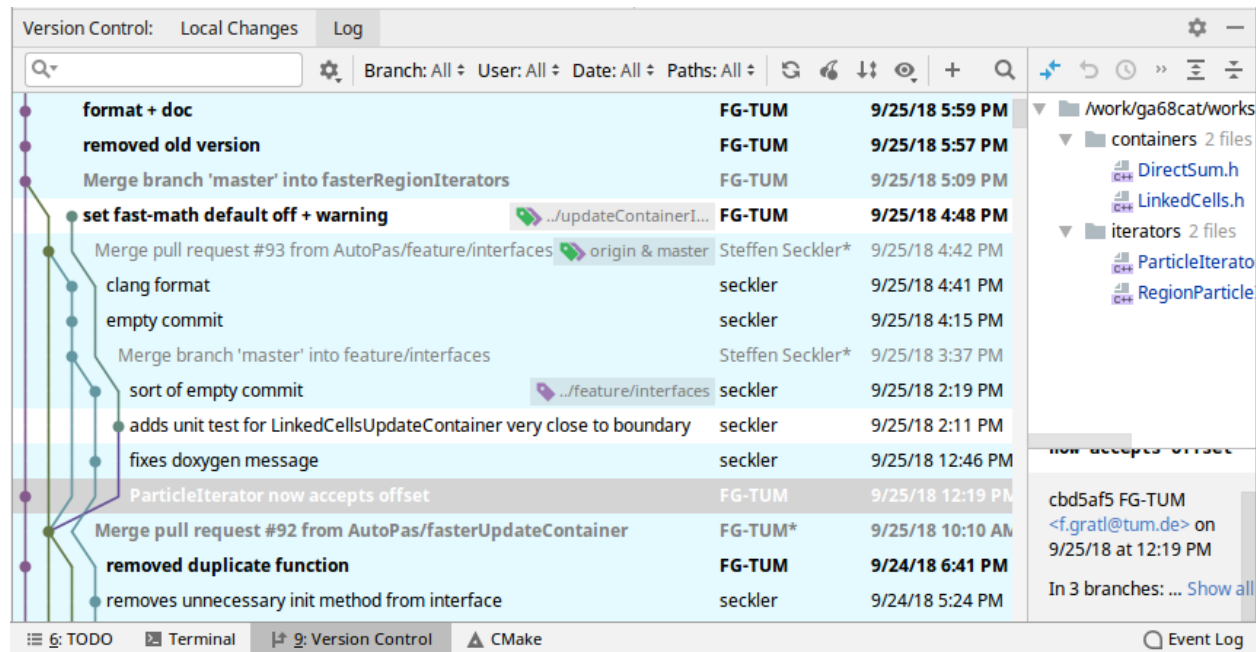
What is git?

- Officially:
git - the stupid content tracker
- Distributed Version Control System (DVCS)
⇒ local + remote repository!
- git-scm.com



Getting help

- Manpages: `man git`, `man gittutorial`, `man giteveryday`
- Built-in help: `git help $COMMAND` (same as `man git-$COMMAND`)
- GUI tools for all platforms exist (but not covered here).
- IDEs can handle git (eclipse, intelliJ, CLion...)



A new repository (GitHub)

- Create a new repository on GitHub or look for one you like.

Search or jump to... Pull requests Issues Marketplace Explore

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner **Repository name**

FG-TUM / DemoRepo ✓

Great repository names are short and memorable. Need inspiration? How about **solid-sniffle**.

Description (optional)

This is not a useful description!

☒ **Public**
Anyone can see this repository. You choose who can commit.

☐ **Private**
You choose who can see and commit to this repository.

☒ **Initialize this repository with a README**
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None** | Add a license: **None** ⓘ

Create repository

A new repository (GitLab)

- Create a new repository on GitLab or look for one you like.

The screenshot shows the 'New project' form in the GitLab web interface. The top navigation bar includes the 'lrz' logo, links for 'Projects', 'Groups', 'Activity', 'Milestones', and 'Snippets', a search bar, and user profile icons. The left sidebar contains a 'New project' section with explanatory text and a tip. The main form area has three tabs: 'Blank project' (selected), 'Create from template', and 'Import project'. The 'Blank project' tab contains fields for 'Project path' (set to 'https://gitlab.lrz.de/' and 'ga68cat'), 'Project name' (set to 'DemoRepo'), and a 'Project description (optional)' field with a placeholder message. Below these are 'Visibility Level' options: 'Private', 'Internal', and 'Public' (selected). At the bottom, there is a checked checkbox for 'Initialize repository with a README' and two buttons: 'Create project' and 'Cancel'.

Projects

New project

A project is where you house your files (repository), plan your work (issues), and publish your documentation (wiki), [among other things](#).

All features are enabled for blank projects, from templates, or when importing, but you can disable them afterward in the project settings.

Does your project include binary files, images and/or logs?

Use [Git Large File Storage](#) to manage those. Read our [Terms of Usage](#) on that topic.

Tip: You can also create a project from the command line. [Show command](#)

Blank project Create from template Import project

Project path

Project name

Want to house several dependent projects under the same namespace? [Create a group](#)

Project description (optional)

Visibility Level

☐ Private
Project access must be granted explicitly to each user.

☐ Internal
The project can be accessed by any logged in user.

☒ Public
The project can be accessed without any authentication.

☒ **Initialize repository with a README**
Allows you to immediately clone this project's repository. Skip this if you plan to push up an existing repository.

Create project **Cancel**

Bring it on your machine (GitHub)

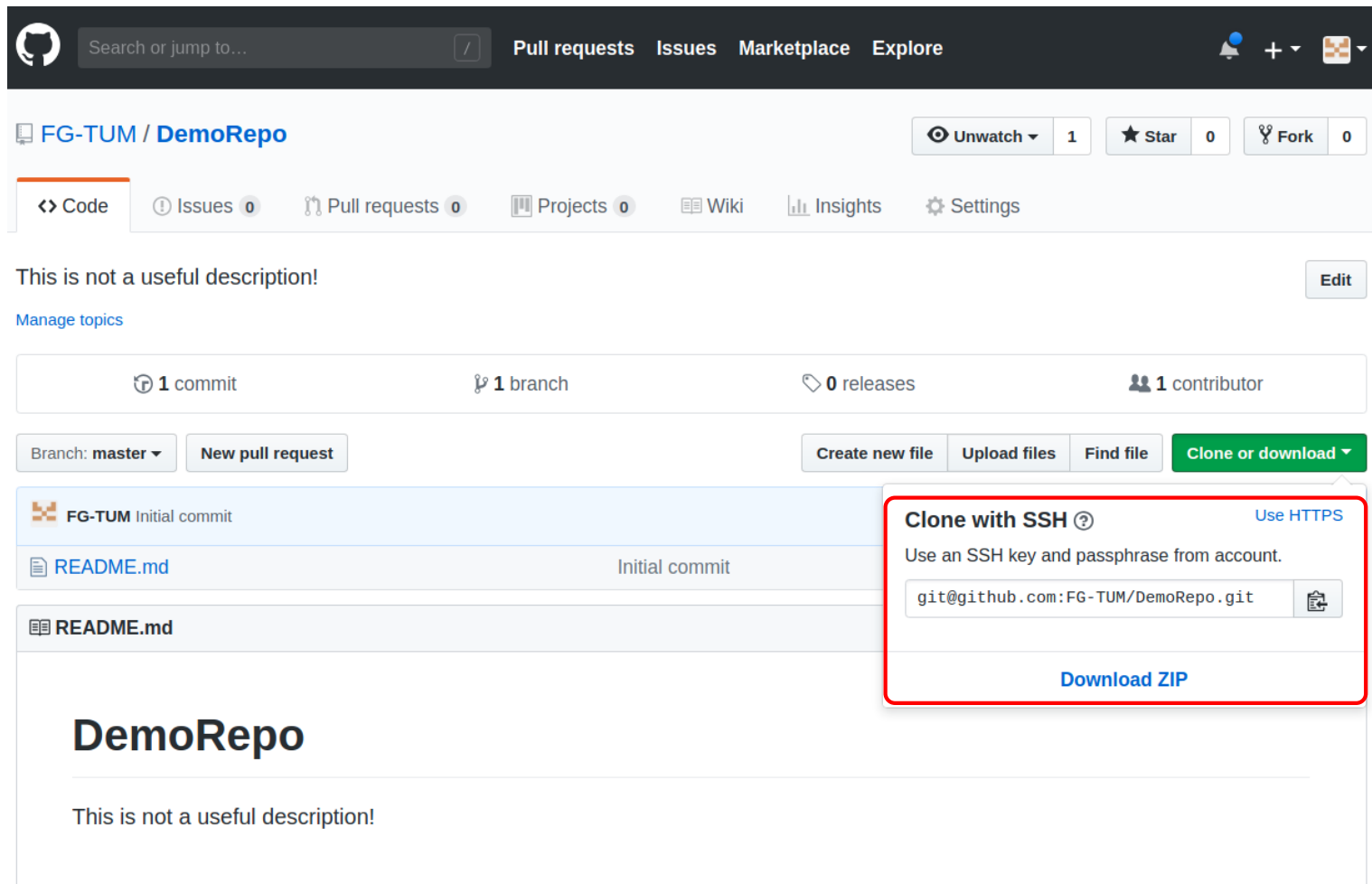
- Clone via **ssh** (requires public-key upload¹) or https.

The screenshot shows the GitHub interface for the repository 'FG-TUM / DemoRepo'. The repository has 1 commit, 1 branch, 0 releases, and 1 contributor. The 'Clone or download' button is highlighted with a red box. A modal is open, showing the 'Clone with SSH' option, which includes the repository URL 'git@github.com:FG-TUM/DemoRepo.git' and a 'Download ZIP' button.

¹<https://help.github.com/articles/connecting-to-github-with-ssh/>

Bring it on your machine (GitHub)

- Clone via **ssh** (requires public-key upload¹) or https.



Search or jump to... Pull requests Issues Marketplace Explore

FG-TUM / DemoRepo

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

This is not a useful description! Edit

Manage topics

1 commit 1 branch 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find file Clone or download

FG-TUM Initial commit

README.md Initial commit

README.md

DemoRepo

This is not a useful description!

Clone with SSH Use HTTPS

Use an SSH key and passphrase from account.

git@github.com:FG-TUM/DemoRepo.git

Download ZIP

¹<https://help.github.com/articles/connecting-to-github-with-ssh/>

Bring it on your machine (GitLab)

- Clone via **ssh** (requires public-key upload²) or https.

The screenshot shows the GitLab web interface for a repository named 'DemoRepo'. The left sidebar contains navigation links: Project, Details, Activity, Cycle Analytics, Repository, Issues (0), Merge Requests (0), Wiki, and Settings. The main content area shows the repository details, including the name 'DemoRepo', a description 'This is not a useful description!', and the Project ID '25304'. Below this, there are buttons for Star (0), Fork (0), and a clone URL dropdown menu. The clone URL is 'SSH git@gitlab.lrz.de:ga68cat/Demo'. A red box highlights the clone URL and the 'Copy URL to clipboard' button. Below the clone URL, there are buttons for 'Add Changelog', 'Add License', 'Add Contribution guide', and 'Set up CI/CD'. At the bottom, there is a table showing the commit history.

Name	Last commit	Last update
README.md	Initial commit	42 seconds ago

²<https://docs.gitlab.com/ee/ssh/>

Bring it on your machine

git clone <repository> [<directory >]

- Copies remote repository to local machine.
- Fetches all branches.
- Requires target folder to be empty!

Example:

```
1 ~$ git clone git@gitlab.lrz.de:ga68cat/DemoRepo.git
2 Cloning into 'DemoRepo'...
3 ~$ cd DemoRepo
4 ~/DemoRepo$ ls -la
5 . .. .git README.md
```

Gaining Overview

git status

- Show status of current working copy.
- List modifications, new files, deletions, merge conflicts...
- Always provides hints what to do! (not shown here)

git log [-<number>]

- Show information about the last n commits.
- Range based query also possible.

Example:

```
1 ~/DemoRepo$ git status
2 On branch master
3 Your branch is up—to—date with 'origin/master'.
4 nothing to commit, working directory clean
5 ~/DemoRepo$ git log -1
6 commit 139e2f8be08bb6fba96b27fd30f31008880584d4
7 Author: FG—TUM <FG—TUM@users.noreply.github.com>
8 Date: Fri Oct 5 11:52:47 2018 +0200
9     Initial commit
```

Ignoring Stuff

.gitignore

- List of files and paths to be ignored by git.
- Accepts "*" as wildcard.
- This file should be uploaded as any other.
- Useful for output or IDE files.

Example:

```
1 ~/DemoRepo$ touch a.foo b.foo
2 ~/DemoRepo$ echo "*.foo" >> .gitignore
3 ~/DemoRepo$ mkdir dir && touch dir/one dir/two dir/c.foo
4 ~/DemoRepo$ echo "dir" >> .gitignore
```

Small Changes

- Modify README.md
- Create a new file FileA.txt

```

1 ~/DemoRepo$ echo "some useless text..." >> README.md
2 ~/DemoRepo$ echo "Let there be text\nCommon line." >> FileA.txt
3 ~/DemoRepo$ ls
4 FileA.txt README.md
5 ~/DemoRepo$ git status
6 On branch master
7 Your branch is up—to—date with 'origin/master'.
8 Changes not staged for commit:
9     modified: README.md
10 Untracked files:
11     FileA.txt
12 no changes added to commit

```

Committing Changes

git add

- Track new files.
- Stage existing files.

git commit [--amend | --message | --signoff]

- Commit changes to local repository.
- Checkpoint to revert or compare to.

Example:

```

1 ~/DemoRepo$ git add FileA.txt README.md
2 ~/DemoRepo$ git status
3 On branch master
4 Your branch is up—to—date with 'origin/master'.
5 Changes to be committed:
6     new file:   FileA.txt
7     modified:  README.md
8 ~/DemoRepo$ git commit --message "meaningful message"
9 ~/DemoRepo$ git status
10 On branch master
11 Your branch is ahead of 'origin/master' by 1 commit.
```

Pushing Changes

git push

- Sends all committed changes to remote.

Example:

```
1 ~/DemoRepo$ git push
2 To git@gitlab.lrz.de:ga68cat/DemoRepo.git
3 139e2f8..014affb master -> master
```

Pulling Changes

git pull [--rebase]

- Updates all files in the local branch.
- Updates information about other remote branches.
- rebase: Put local changes on top of remote instead of merging.

Example: (Suppose someone added a line break in README.md.)

```

1 ~/DemoRepo$ git pull
2 From git@gitlab.lrz.de:ga68cat/DemoRepo.git
3 014affb..d20ae23 master -> origin/master
4 Updating 014affb..d20ae23
5 Fast-forward
6 README.md | 1 +
7 1 file changed, 1 insertion(+)
  
```

Starting a new Feature

git checkout <branch> | <File>

- Switch to existing branch.
- OR reset <File> to last commit.

git branch [<new_branch>]

- Create a new branch.
- OR list all local branches.

Example:

```
1 ~/DemoRepo$ git branch branchForAwesomeFeature
2 ~/DemoRepo$ git branch
3 branchForAwesomeFeature
4 * master
5 ~/DemoRepo$ git checkout branchForAwesomeFeature
6 Switched to branch 'branchForAwesomeFeature'
```

Merging branches

git merge <branch>

- Merges given branch in current branch.

Example: (Suppose we changed FileA.txt on the new branch and someone else on master.)

```

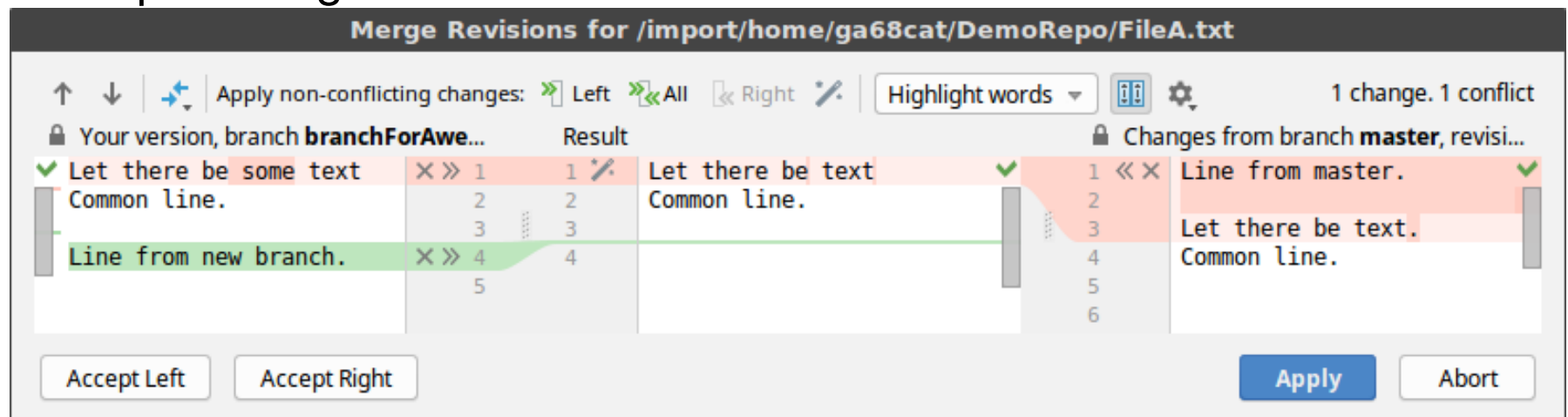
1 ~/DemoRepo$ git merge master
2 Auto—merging FileA.txt
3 CONFLICT (content): Merge conflict in FileA.txt
4 Automatic merge failed; fix conflicts and then commit the result.
5 ~/DemoRepo$ git status
6 On branch branchForAwesomeFeature
7 You have unmerged paths.
8 Unmerged paths:
9   both modified: FileA.txt
  
```

Resolving conflicts

git mergetool (needs to be configured)

- vimdiff, Meld, SmartGit...
- IDEs

Example: Merge view in CLion:

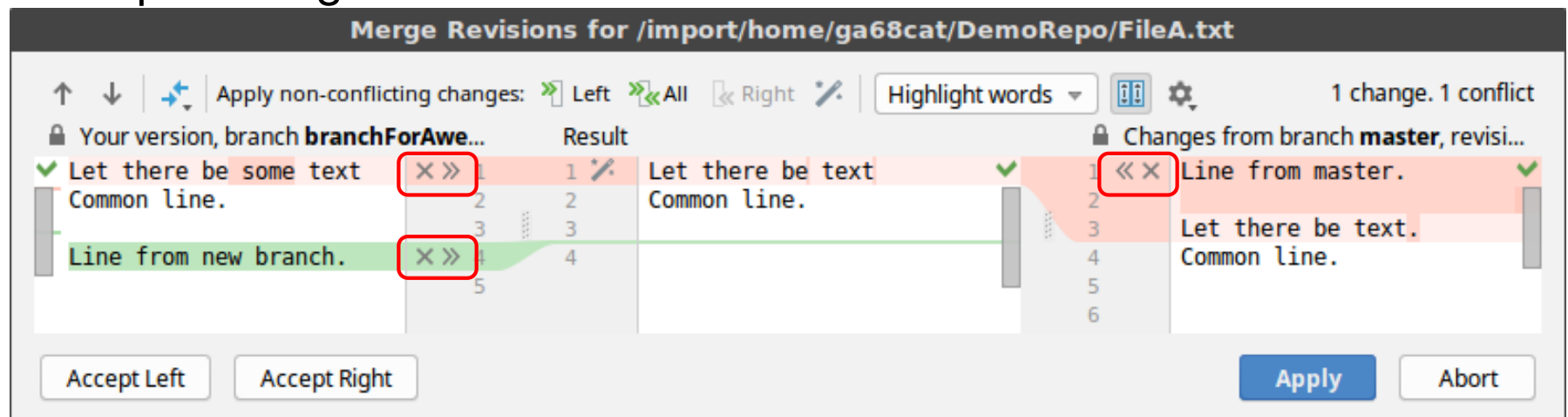


Resolving conflicts

git mergetool (needs to be configured)

- vimdiff, Meld, SmartGit...
- IDEs

Example: Merge view in CLion:

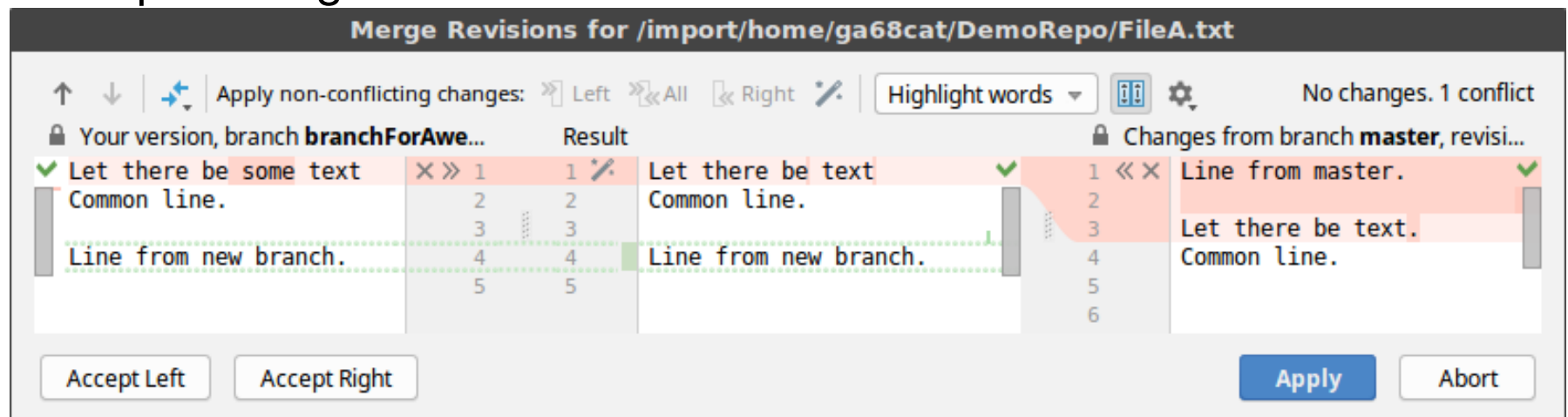


Resolving conflicts

git mergetool (needs to be configured)

- vimdiff, Meld, SmartGit...
- IDEs

Example: Merge view in CLion:

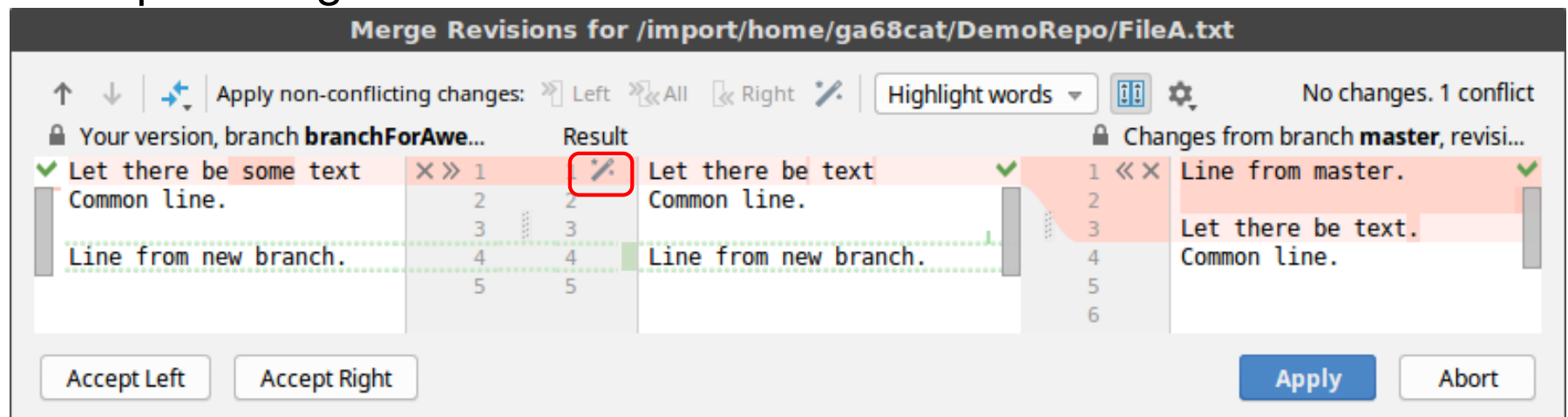


Resolving conflicts

git mergetool (needs to be configured)

- vimdiff, Meld, SmartGit...
- IDEs

Example: Merge view in CLion:

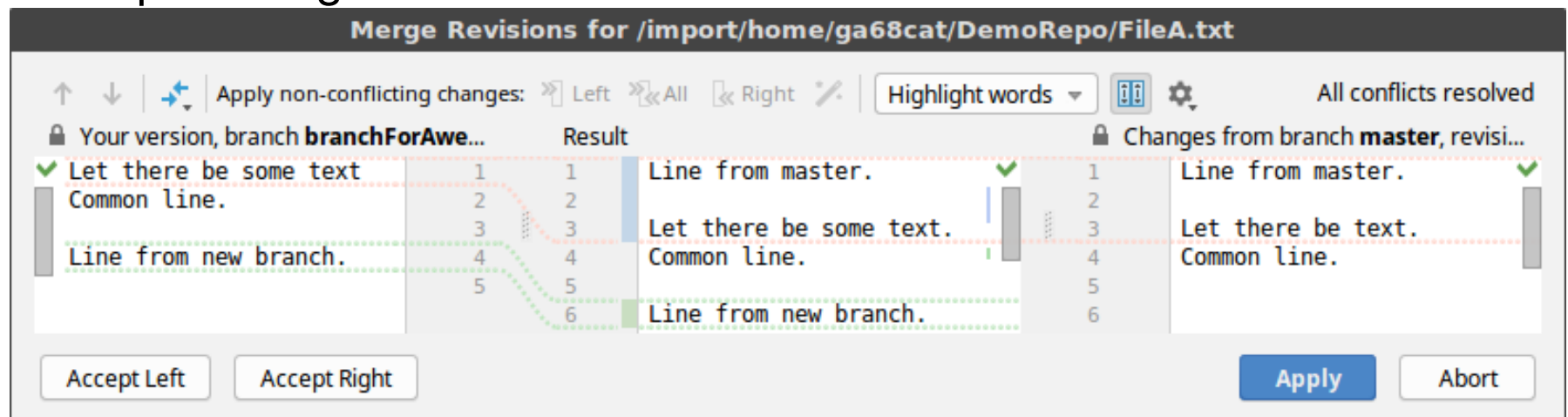


Resolving conflicts

git mergetool (needs to be configured)

- vimdiff, Meld, SmartGit...
- IDEs

Example: Merge view in CLion:

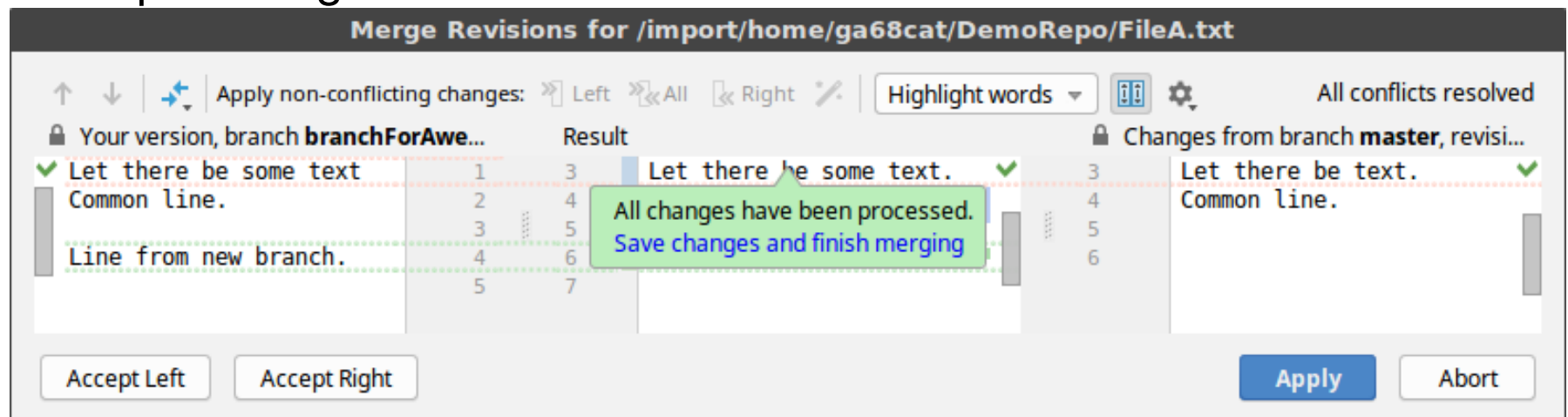


Resolving conflicts

git mergetool (needs to be configured)

- vimdiff, Meld, SmartGit...
- IDEs
- After resolving all conflicts: **git add** and **git commit**.

Example: Merge view in CLion:



Merging back to master (GitHub)

Search or jump to... Pull requests Issues Marketplace Explore

FG-TUM / DemoRepo Unwatch 1 Star 0 Fork 0

Code Issues 1 Pull requests 1 Projects 0 Wiki Insights Settings

Branch for awesome feature #2

Open FG-TUM wants to merge 2 commits into master from branchForAwesomeFeature

Conversation 0 Commits 2 Checks 0 Files changed 1 +2 -1

FG-TUM commented 2 minutes ago • edited Owner + ...

- ☐ comment with tickbox
- ☒ comments like this directly close #1 (<-Issue)

FG-TUM added some commits 3 hours ago

- Who knows... e2ca5ab
- merged from master Verified 3095cce

Add more commits by pushing to the **branchForAwesomeFeature** branch on **FG-TUM/DemoRepo**.

Continuous integration has not been set up
Several apps are available to automatically catch bugs and enforce style.

This branch has no conflicts with the base branch
Merging can be performed automatically.

Merge pull request or view [command line instructions](#).

Reviews
No reviews

Assignees
No one—assign yourself

Labels
None yet

Projects
None yet

Milestone
No milestone

Notifications
Unsubscribe
You're receiving notifications because you authored the thread.

Merging back to master (GitHub)

Search or jump to... Pull requests Issues Marketplace Explore

FG-TUM / DemoRepo Unwatch 1 Star 0 Fork 0

Code Issues 1 Pull requests 1 Projects 0 Wiki Insights Settings

Branch for awesome feature #2

Open FG-TUM wants to merge 2 commits into master from branchForAwesomeFeature

Conversation 0 Commits 2 Checks 0 Files changed 1 +2 -1

FG-TUM commented 2 minutes ago • edited

- ☐ comment with tickbox
- ☒ comments like this directly close #1 (<-Issue)

FG-TUM added some commits 3 hours ago

- Who knows... e2ca5ab
- merged from master 3095cce (Verified)

Add more commits by pushing to the branchForAwesomeFeature branch on FG-TUM/DemoRepo.

Continuous integration has not been set up
Several apps are available to automatically catch bugs and enforce style.

This branch has no conflicts with the base branch
Merging can be performed automatically.

Merge pull request or view command line instructions.

Reviewers: No reviews

Assignees: No one—assign yourself

Labels: None yet

Projects: None yet

Milestone: No milestone

Notifications: Unsubscribe

You're receiving notifications because you authored the thread.

Merging back to master (GitLab)

The screenshot displays the GitLab web interface for a repository named 'DemoRepo'. The left sidebar contains navigation links: Project, Repository, Issues (with a count of 1), Merge Requests (highlighted with a red box and a count of 1), Wiki, and Settings. The main content area shows a merge request titled 'Branch for awesome feature' opened 3 minutes ago by 'FG-TUM'. It includes a 'Request to merge' section with options to 'Open in Web IDE' and 'Check out branch'. Below this, there are checkboxes for 'Merge', 'Remove source branch', and 'Squash commits', along with a 'Modify commit message' button. The merge request also shows it 'Closes #1' and has a link to 'Assign yourself to this issue'. At the bottom, there are tabs for 'Discussion' (0), 'Commits' (2), and 'Changes' (0), with a timestamp of '10 Oct, 2018 2 commits'. The right sidebar contains a 'Todo' section with fields for Assignee, Milestone, Time tracking, Labels, Lock merge request, and Notifications, along with a reference to 'ga68cat/DemoRepo!1'.

lrz Projects Groups Activity Milestones Snippets Search or jump to...

DemoRepo

Project

Repository

Issues 1

Merge Requests 1

Wiki

Settings

FG-TUM > DemoRepo > Merge Requests > 11

Open Opened 3 minutes ago by FG-TUM 1 of 2 tasks completed Edit Close merge request

Branch for awesome feature

☐ comment with tickboxes

☒ comments like this directly close #1 (<-Issue)

Edited 2 minutes ago by FG-TUM

Request to merge branchForAwesomeFeature into master Open in Web IDE Check out branch

☒ Merge ☐ Remove source branch ☐ Squash commits

Closes #1

[Assign yourself to this issue](#)

You can merge this merge request manually using the [command line](#)

0 0

Discussion 0 Commits 2 Changes 0

10 Oct, 2018 2 commits

Todo Add todo

Assignee Edit No assignee - assign yourself

Milestone Edit None

Time tracking ? No estimate or time spent

Labels Edit None

Lock merge request Edit Unlocked

1 participant

Notifications

Reference: ga68cat/DemoRepo!1

Merging back to master (GitLab)

The screenshot shows a GitLab interface for a project named 'DemoRepo'. The left sidebar contains navigation links: Project, Repository, Issues (1), Merge Requests (1), Wiki, and Settings. The main content area displays a Merge Request titled 'Branch for awesome feature' created by 'FG-TUM' 3 minutes ago. It includes a 'Request to merge' section with options to 'Open in Web IDE' or 'Check out branch'. Below this is a 'Merge' button and checkboxes for 'Remove source branch' and 'Squash commits'. The right sidebar shows a 'Todo' list with 'Assignee' (No assignee - assign yourself), 'Milestone' (None), 'Time tracking' (No estimate or time spent), 'Labels' (None), 'Lock merge request' (Unlocked), '1 participant', and 'Notifications' (enabled). The bottom of the page shows 'Discussion 0', 'Commits 2', and 'Changes 0'.

irz Projects Groups Activity Milestones Snippets Search or jump to...

DemoRepo

Project

Repository

Issues 1

Merge Requests 1

Wiki

Settings

FG-TUM > DemoRepo > Merge Requests > 11

Open Opened 3 minutes ago by FG-TUM 1 of 2 tasks completed Edit Close merge request

Branch for awesome feature

☐ comment with tickboxes

☒ comments like this directly close #1 (<-Issue)

Edited 2 minutes ago by FG-TUM

Request to merge branchForAwesomeFeature into master Open in Web IDE Check out branch

Merge ☐ Remove source branch ☐ Squash commits Modify commit message

Closes #1

Assign yourself to this issue

You can merge this merge request manually using the command line

0 0

Discussion 0 Commits 2 Changes 0

10 Oct, 2018 2 commits

Todo Add todo

Assignee Edit No assignee - assign yourself

Milestone Edit None

Time tracking ? No estimate or time spent

Labels Edit None

Lock merge request Edit Unlocked

1 participant

Notifications

Reference: ga68cat/DemoRepo!1

Summary

- **git pull**
- **git branch** branchForAwesomeFeature
- **git checkout** branchForAwesomeFeature
- Do what you must...
- **git commit**
- **git merge** master
- **git push**
- Create pull request

And anytime you feel lost:

- **git status**

Backup Content

Initialize an existing Repository

git init

- Create a new local repository
(.git folder)

git remote add <name> <url>

- Manage remote tracking repositories.

Example: First create repository on GitLab³

```
1 ~/DemoRepo$ git init
2 Initialized empty Git repository in ~/DemoRepo/.git/
3 ~/DemoRepo$ git add . && git commit -m"First commit"
4 ~/DemoRepo$ git remote add origin git@gitlab.lrz.de:ga68cat/DemoRepo.git
5 ~/DemoRepo$ git remote -v
6 origin git@gitlab.lrz.de:ga68cat/DemoRepo.git (fetch)
7 origin git@gitlab.lrz.de:ga68cat/DemoRepo.git (push)
```

³<https://help.github.com/articles/adding-an-existing-project-to-github-using-the-command-line/>

Undoing a Commit

git revert

[--soft | --hard] [<commit>]

- Undo commit (not pushed yet!).
- soft: preserve changes.
- hard: revert changes.

Example:

```
1 ~/DemoRepo$ git reset --soft HEAD~1
2 ~/DemoRepo$ git status
3 On branch master
4 Your branch is up—to—date with 'origin/master'.
5 Changes to be committed:
6   new file:   FileA.txt
7   modified:  README.md
8 ~/DemoRepo$ git reset HEAD README.md
9 Unstaged changes after reset:
10 M README.md
```

[<tree-ish>] <paths>...

- tree-ish: can be a commit or relative to HEAD
- Unstaging single files (not committed yet!).

Who did that!?

git blame [-w] <File>

- Show author of last line change.
- -w Exclude whitespace-only changes

Example: (Suppose there is a second author)

```
1 ~/AutoPas$ git blame README.md
2 f5344b80 README.md (FG-TUM 2018-07-31 15:53:56 +0200 1) # AutoPas
3 e46ec677 README.md (Seckler 2018-04-13 17:03:54 +0900 2) AutoPas is a ...
```

CLion: Annotations



Put stuff aside

git stash [save <message> | list | pop [<stash>]]

- Save all current changes with a message.
- List all stashes.
- Apply a stash and remove it from the list.

Example:

```

1 ~/DemoRepo$ git checkout branchForAwesomeFeature
2 error: Your local changes to the following files would be overwritten by checkout:
3     README.md
4 Please commit your changes or stash them before you switch branches.
5 Aborting
6 ~/DemoRepo$ git stash save "my message"
7 Saved working directory and index state On master: my message
8 ~/DemoRepo$ git stash list
9 stash@{0}: On master: my message
10 ~/DemoRepo$ git stash pop
11 Changes not staged for commit:
12     README.md
  
```